

INTRODUCTION

The Dynamic Neural Simulator (DNSim) is a Matlab-based tool that simplifies the process of building and sharing large differential equation models. It was motivated by challenges encountered studying multilayer networks of multicompartment biophysical spiking neurons. Major features include:

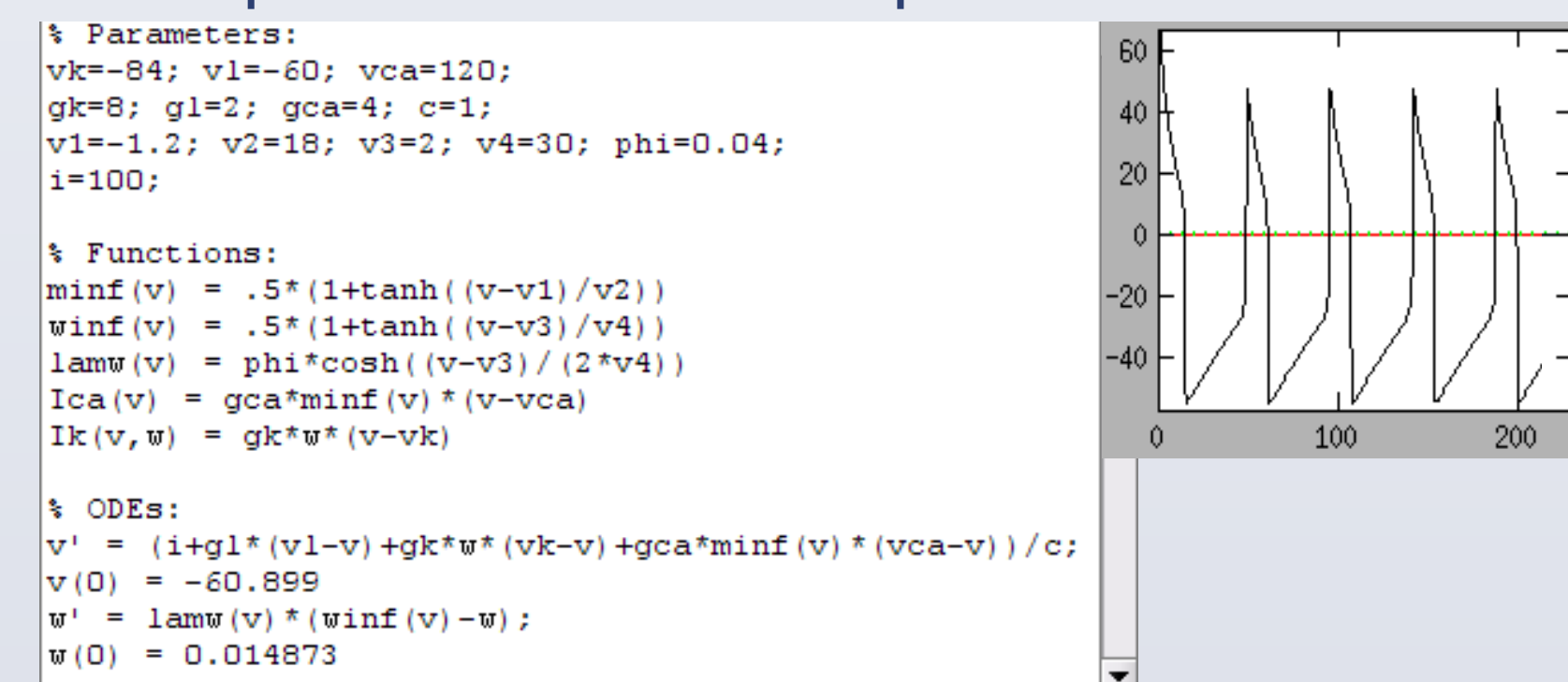
- Dynamic simulation:** *interactively tune parameters during simulation*
- InfiniteBrain.org:** *download/upload existing models from online repository*
- Batches and cluster computing:** *use a cluster to explore search space*
- Model remixing:** *combine select components from two or more models*
- Model comparison:** *list all differences between two models*
- Model notes and reversion:** *revert back to earlier models with notes*

Models can be seen at infinitebrain.org/models. Install instructions for DNSim can be found at infinitebrain.org/setup.

Model Building

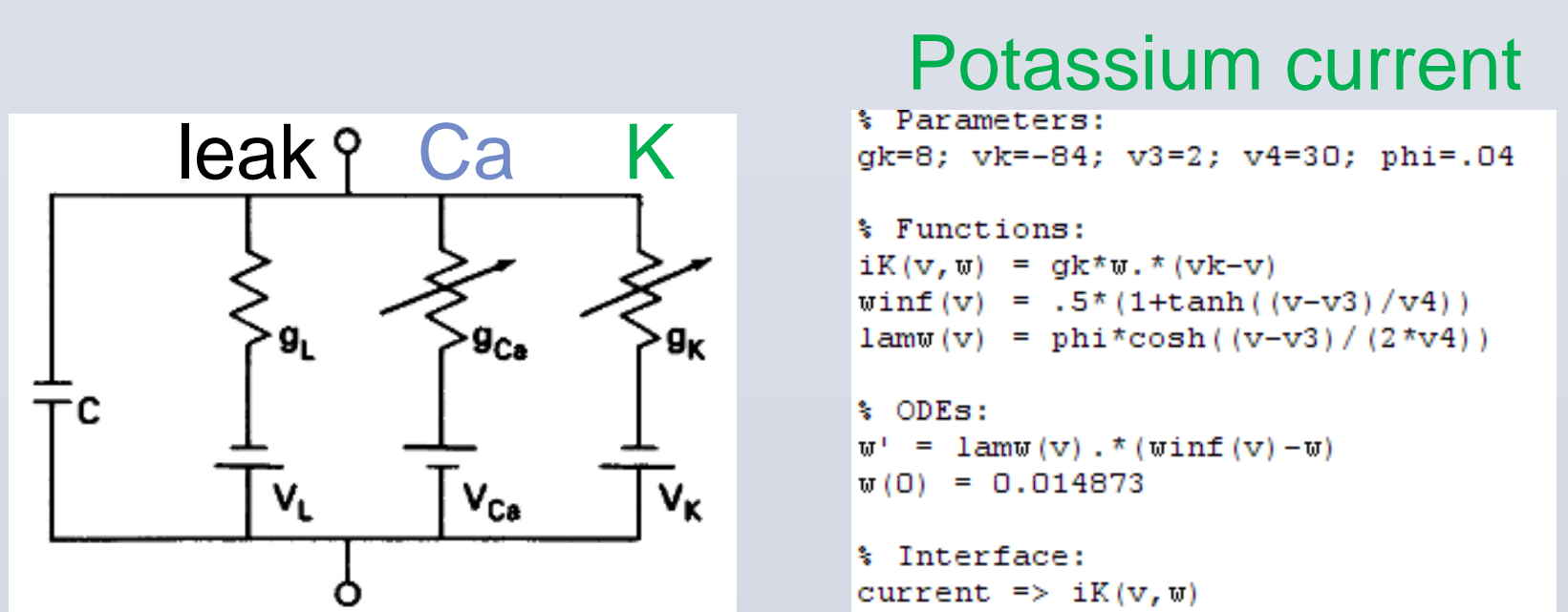
Approach 1: Write simple equations.

Example: Morris-Lecar equations



Approach 2: Define detailed mechanisms.

Example: Equivalent Morris-Lecar model



Mechanisms: Ca, K, leak

Dynamics: $v' = \text{current}/c$

Full system generated automatically by string substitution

Calcium current

```
% Parameters:
vca=120; gca=4; v1=-1.2; v2=18

% Functions:
minf(v) = .5*(1+tanh((v-v1)/v2))
iCa(v) = gca*minf(v).*(vca-v)

% Interface:
current => iCa(v)
```

GRAPHICAL USER INTERFACE

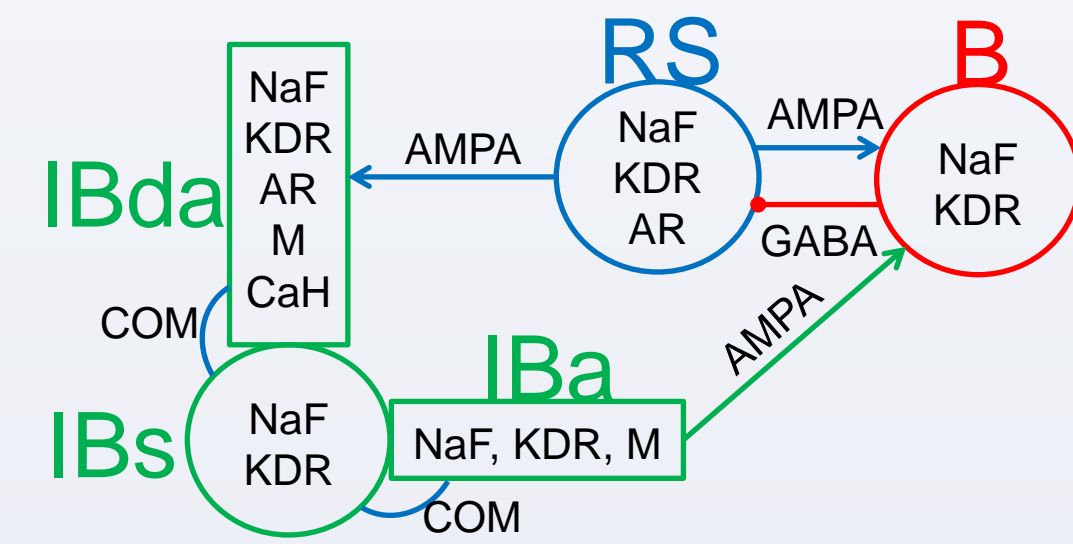
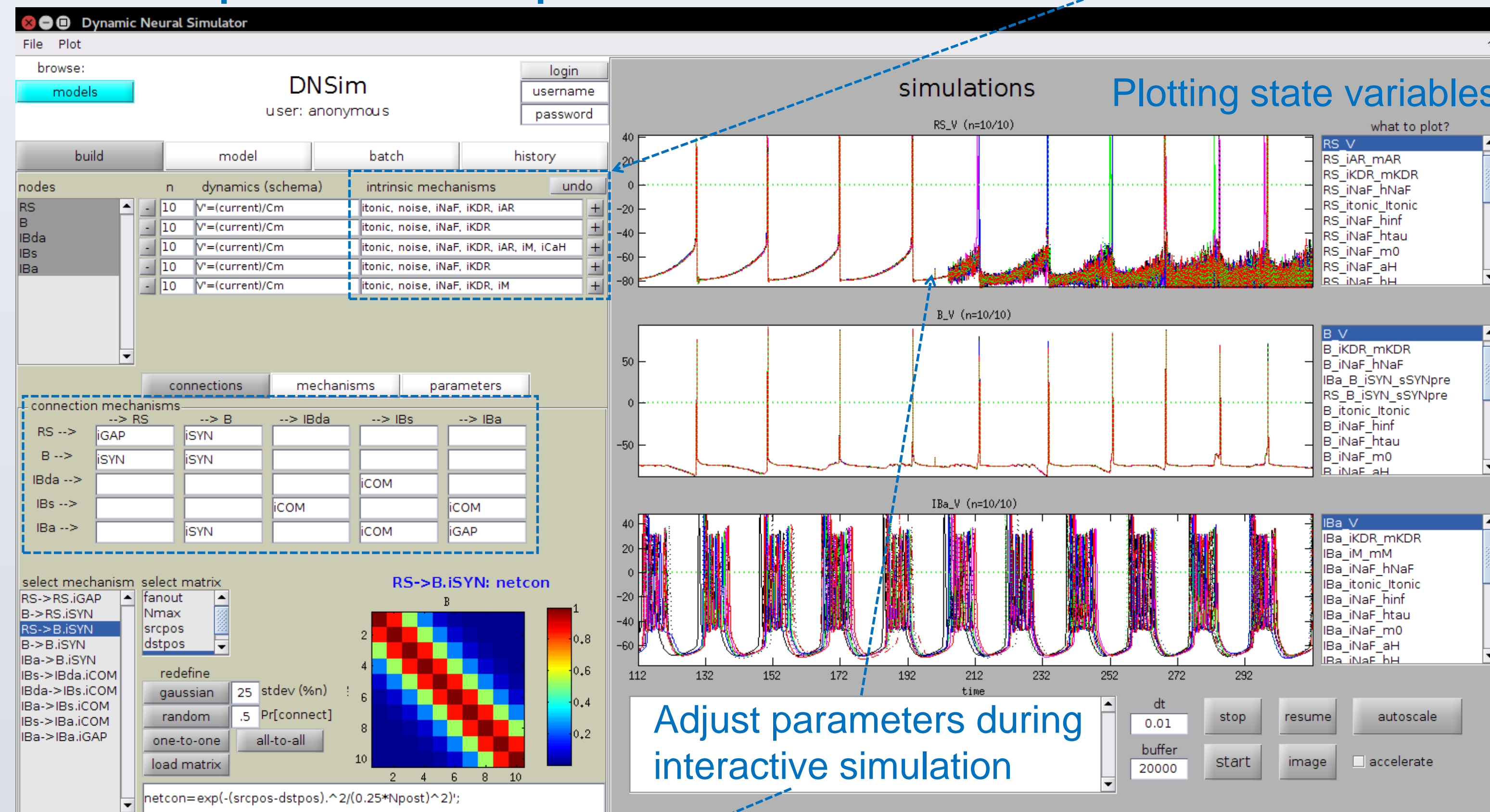


Figure. Interlaminar cortical microcircuit model. Excitatory regular spiking (RS), inhibitory basket (B), and bursting (IB) cells are modeled as biophysical spiking neurons; the IB model has three compartments: apical dendrite (IBda), soma (IBs), and axon (IBa). IB compartments are connected by an ohmic mechanism (COM); RS and B synapses are AMPAergic and GABAergic, respectively. In network simulation (i.e., $n>1$), gap junctions are added between adjacent RS cells and between IB axons.

Building a highly-detailed model from existing mechanisms is as easy as writing lists.

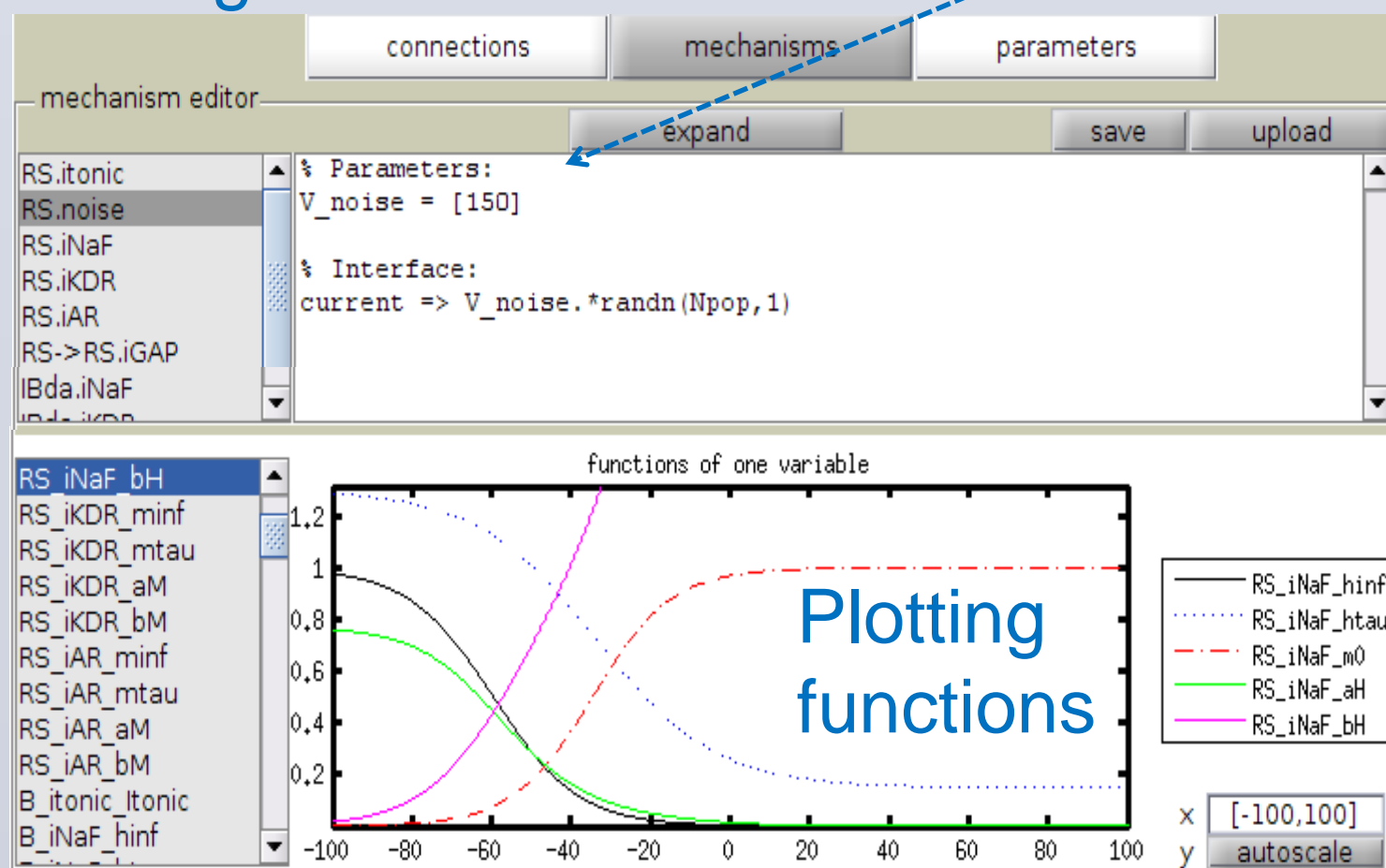
DNSim simplifies common operations:



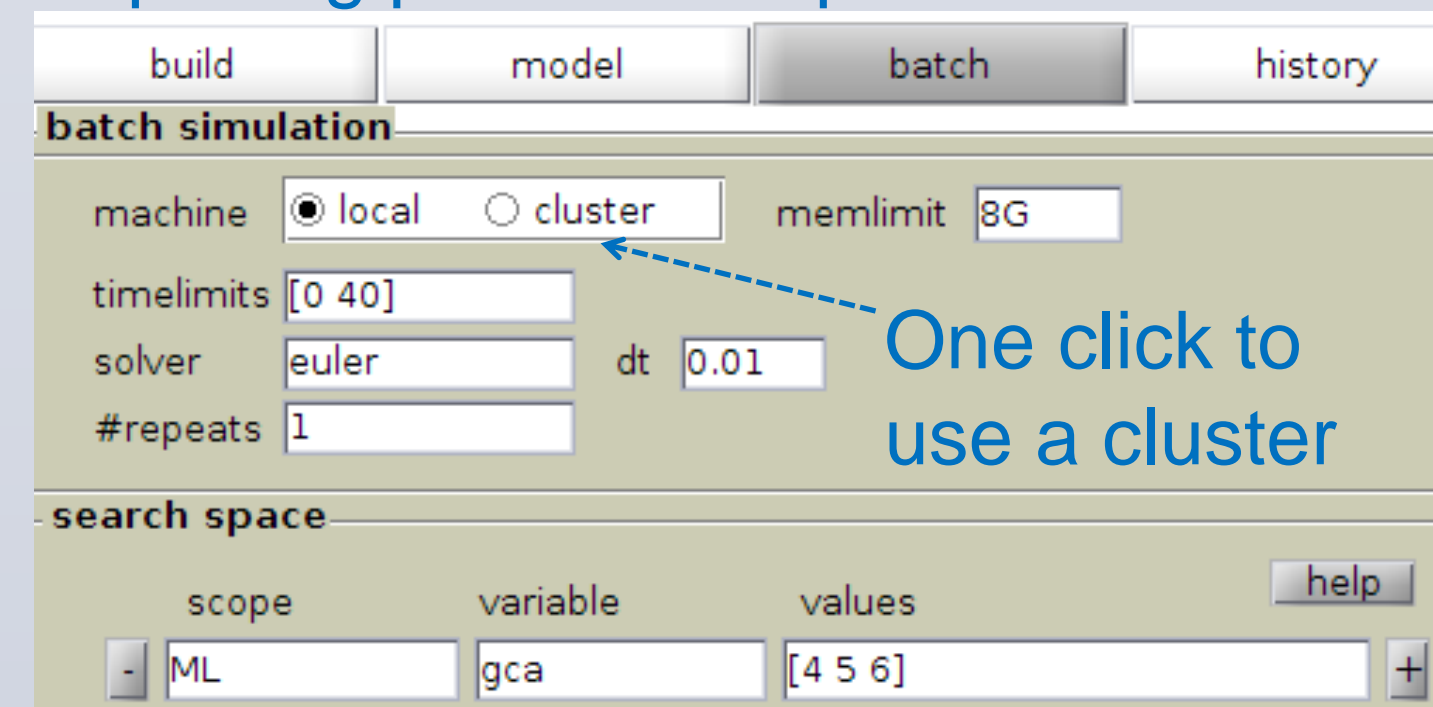
Adjust parameters during interactive simulation

Changing network connectivity

Editing mechanisms



Exploring parameter space

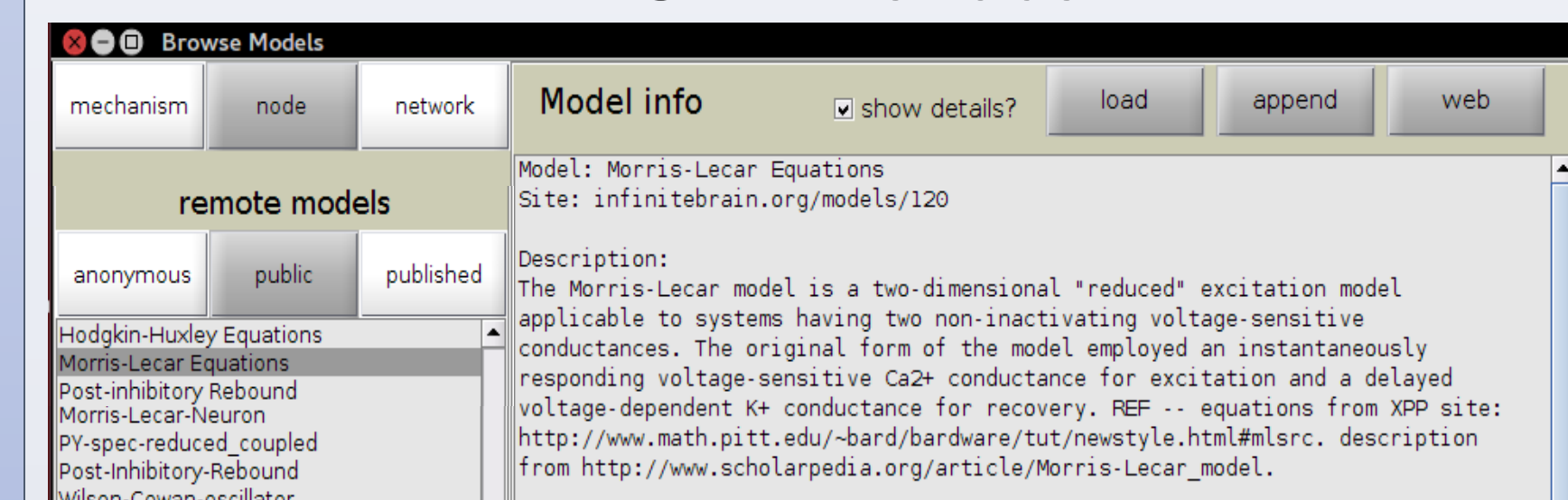


- Batch management - DNSim can run many simulations varying some aspect of the model specified using a simple syntax.
- Cluster computing - DNSim handles job creation and submission using qsub from a login node.
- Results can be saved to disk with a model file for repeating simulations.

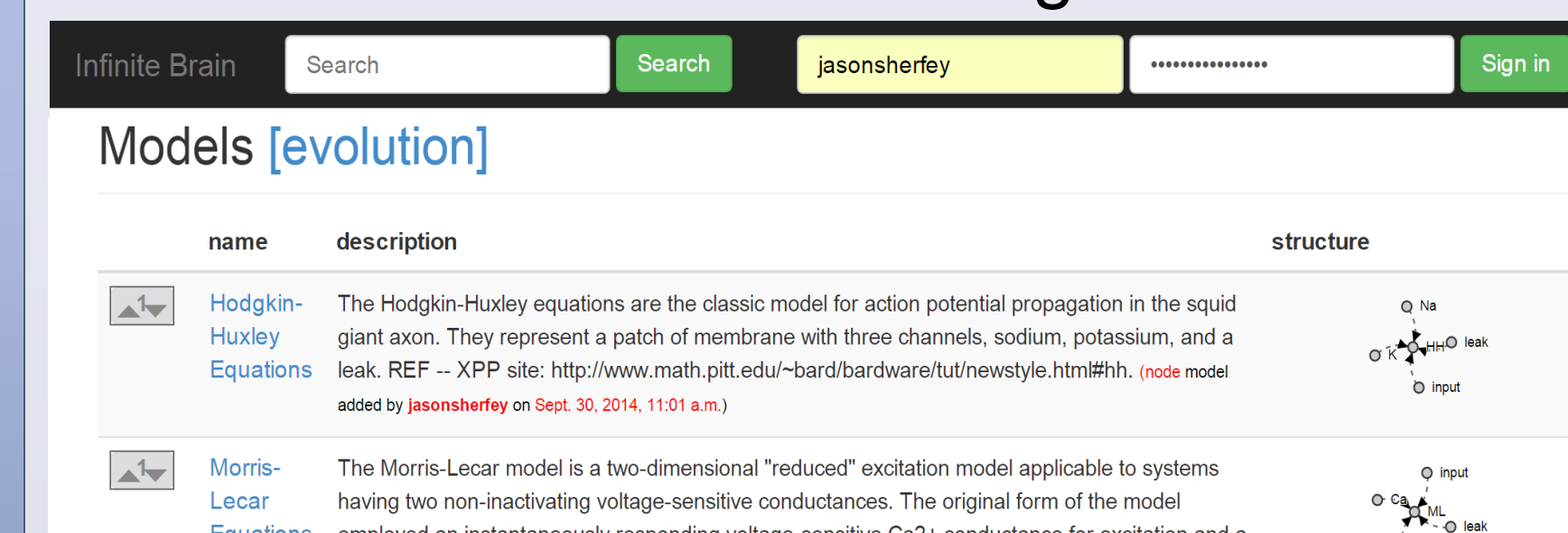
MODEL REPOSITORY

Models can be stored locally or in an online database accessible from the DNSim GUI itself or a web browser. Every online model has its own detail page to aid sharing models and having public discussions. Privacy settings enable users to control who can see their models. Citations can be added to models, and all models with citations appear in a "published" model list.

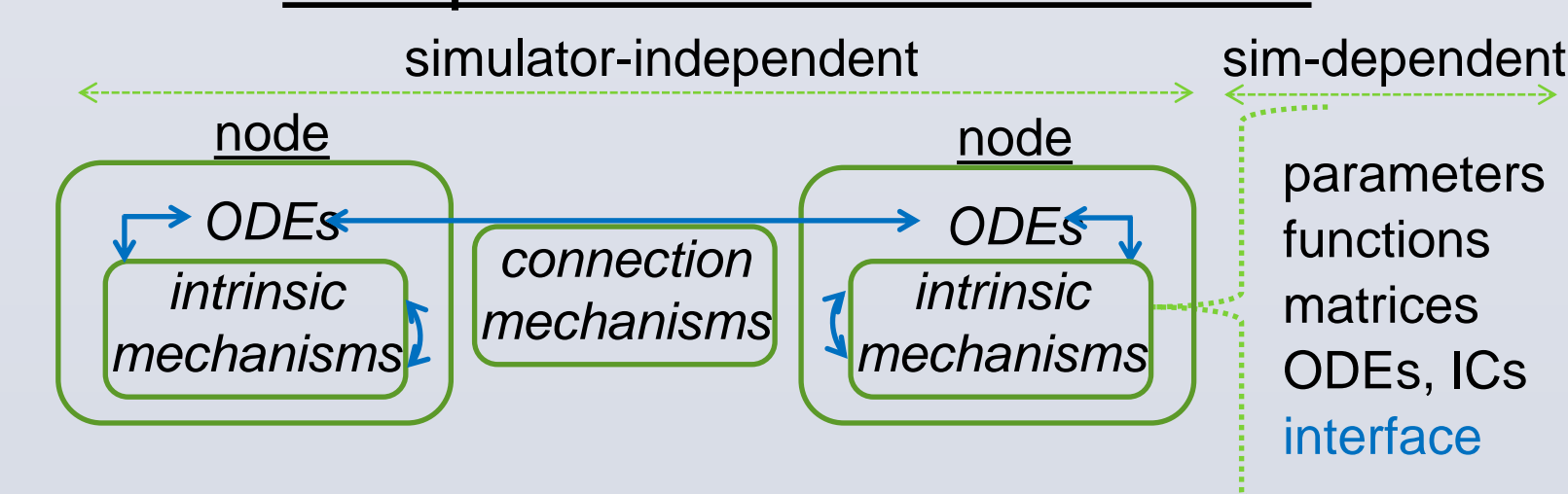
DNSim Interface



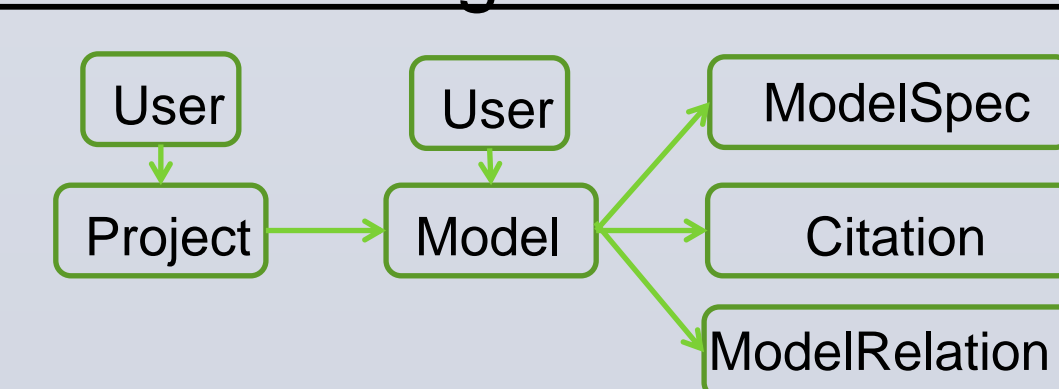
InfiniteBrain.org



Components of DNSim model



InfiniteBrain.org database entities



Limitations of DNSim: (1) lack of explicit spatial representation (limits the spatial complexity of models that can be easily implemented); (2) lack of unit handling (requires users ensure consistency).

Acknowledgments

This work was partially supported by grants NIH-T90DA32484 to JS and NSF DMS-1042134 to NK.